What is claimed is:

1. A method of correcting an audio level of a stored program asset, comprising:

retrieving a stored program asset, the asset having audio encoded at a first loudness setting;

identifying dialog of the audio of the asset;

determining a loudness of the dialog;

comparing the determined loudness to the first loudness setting; and

re-encoding the asset at a second loudness setting corresponding to the second loudness, if the first loudness setting and the determined loudness are different by more than a predetermined amount.

2. The method of claim 1, wherein the audio is encoded at a DIALNORM setting, the method comprising:

determining a loudness of the dialog, wherein the determined loudness is a DIALNORM of the dialog.

3. The method of claim 1, comprising identifying the dialog by:

dividing the audio into time intervals;

determining a loudness of each time interval; and

identifying time intervals with intermediate loudnesses.

4. The method of claim 3, comprising:

determining the loudness of each time interval based on psycho-acoustic criteria.

5. The method of claim 4, comprising:

    determining the loudness of each time interval based on Leq (A).

6. The method of claim 3, further comprising:

    discarding time intervals with high and low loudnesses.

7. The method of claim 3, comprising:

    identifying time intervals with intermediate loudnesses by creating a histogram of the loudnesses of the intervals.

8. The method of claim 3, further comprising:

    determining a loudness of the time intervals having an intermediate loudness.

9. The method of claim 8, comprising determining the loudness of the time intervals having intermediate loudnesses by:

    computing a function of the loudnesses of the time intervals having intermediate loudness.

10. The method of claim 9, wherein:

    the function is an average, a mean or a median of the loudnesses of the time intervals having intermediate loudness.

11. The method of claim 8, wherein determining the loudness of the time intervals having intermediate loudnesses comprises:

    determining a DIALNORM of the time intervals having intermediate loudnesses.

12.   The method of claim 1, further comprising:

correcting compression of the audio of the program.

13.   The method of claim 1, further comprising, prior to retrieving the stored

program asset:

receiving a program from a source; and

storing the program in memory as an asset for later transmission.

14.   The method of claim 1, further comprising:

demultiplexing the audio from the program asset, prior to identifying the

dialog.

15.   The method of claim 1, further comprising:

decompressing the audio, prior to identifying the dialog.

16.   The method of claim 15, further comprising:

decompressing the audio by converting the audio to a pulse coded

modulation format.

17.   The method of claim 1, further comprising:

performing automatic gain control on the audio, prior to identifying the

dialog.

18.   The method of claim 1, further comprising:

filtering the audio, prior to identifying the dialog.

19.   The method of claim 1, comprising identifying the dialog by:

filtering the audio.

20.   The method of claim 19, comprising:

filtering the audio outside of a range of from about 100 Hertz to about

1,000 Hertz.

21.   The method of claim 1, further comprising:

retrieving a second stored program asset, the second asset comprising

audio encoded at a third loudness;

identifying dialog of the asset;

determining a fourth stored loudness of the dialog;

comparing the fourth loudness to the third loudness; and

re-encoding the asset at the fourth loudness, if the third loudness and the

fourth loudness are different by more than a predetermined amount.

22.   The method of claim 1, comprising retrieving an asset comprising a

program, a chapter of a program, or an advertisement.

23.   The method of claim 1, further comprising:

storing the asset with the re-encoded loudness setting.

24.   The method of claim 1, wherein the audio is encoded at a normalized

loudness setting, the method further comprising:

normalizing the determined loudness of the dialog;

comparing the normalized determined loudness to the normalized loudness

setting; and

re-encoding the asset at a second loudness setting corresponding to the

normalized determined loudness if the first loudness setting and the normalized determined

loudness are different by more than a predetermined amount.

25.　A method of correcting an audio level of a stored program asset,

comprising:

retrieving a stored program asset, the asset comprising audio having an

encoded DIALNORM setting;

demultiplexing the audio from the retrieved asset;

decompressing the audio;

identifying dialog of the audio;

determining a DIALNORM of the dialog;

comparing the determined DIALNORM to the encoded DIALNORM

setting;

re-encoding the asset at the determined DIALNORM if the encoded

DIALNORM and the determined DIALNORM are different by more than a predetermined

amount; and

storing the asset with the re-encoded DIALNORM.

26.　The method of claim 25, comprising identifying dialog by:

dividing the audio into time intervals;

determining a loudness of each time interval; and

identifying time intervals with intermediate loudnesses.

27. The method of claim 26, comprising:

identifying time intervals with high, intermediate and low loudnesses by creating a histogram.

28. The method of claim 25, comprising:

decompressing the audio by converting the audio into a pulse code modulation file.

29. The method of claim 28, further comprising:

performing automatic gain control on the pulse code modulation file prior to identifying the dialog.

30. The method of claim 25, further comprising:

filtering the audio.

31. The method of claim 25, further comprising:

correcting compression of the audio.

32. A method of processing an audio level of a stored program asset, comprising:

retrieving a stored program asset, the asset having audio encoded at a loudness setting;

identifying dialog of the asset;

determining a loudness of the dialog; and

comparing the determined loudness to the loudness setting.

33. A system for correcting an audio level of a stored program asset, the system comprising:

means for retrieving a stored program asset, the asset having audio encoded at a first loudness setting;

means for identifying dialog of the asset;

means for determining a loudness of the dialog; and

means for re-encoding the asset at a second loudness setting corresponding to the determined loudness, if the first loudness setting and the determined loudness are different by more than a predetermined amount.

34. The system of claim 33, further comprising:

means for storing the asset.

35. A system for correcting an audio level of a stored program asset, the system comprising:

memory to store the program asset, the asset having audio encoded at a first loudness setting; and

a processor coupled to the memory, the processor being programmed to:

retrieve a stored program asset,

identify dialog of the asset;

determine a loudness of the dialog; and

re-encode the asset at a second loudness setting corresponding to the determined loudness, if the first loudness and the second loudness are different by more than a predetermined amount.

36.   The system of claim 35, wherein the audio is encoded at a DIALNORM setting and the processor is programmed to:

determine DIALNORM of the dialog.

37.   The system of claim 36, wherein the processor is programmed to:

determine a DIALNORM of the intervals in the immediate category;

38.   The system of claim 35, wherein the processor is programmed to identify the dialog by:

dividing the audio into time intervals;

determining a loudness of each time interval; and

identifying time intervals with intermediate loudnesses.

39.   The method of claim 38, wherein the processor is programmed to:

determine the loudness of each time interval based on psycho-acoustic criteria.

40.   The system of claim 39, wherein the processor is programmed to:

determine the loudness based on Leq (A).

41.   The system of claim 38, wherein the processor is further programmed to:

discard time intervals with high and low loudnesses.

42.   The system of claim 38, wherein the processor is programmed to:

identify time intervals with intermediate loudnesses by creating a histogram of the loudnesses of the intervals.

43. The system of claim 38, wherein the processor is programmed to:

determine a loudness of the time intervals in the intermediate category.

44. The system of claim 35, wherein the processor is programmed to:

correct compression of the audio of the program.

45. The system of claim 35, wherein the processor is further programmed to:

receive the program from a source;

store the program in memory for later transmission; and

retrieve the program from memory.

46. The system of claim 35, wherein the processor is further programmed to:

demultiplex the audio from the program.

47. The system of claim 35, wherein the processor is further programmed to:

decompress the audio.

48. The system of claim 47, wherein the processor is further programmed to:

decompress the audio by converting the audio to a pulse coded modulation

format.

49. The system of claim 35, wherein the processor is further programmed to:

perform automatic gain control on the audio prior to identifying the first

and second loudness levels.

50. The system of claim 35, wherein the processor is further programmed to:

filter the audio.

51.   The system of claim 35, wherein the processor is further programmed to identify the dialog by:

filtering the audio.

52.   The system of claim 35, wherein the audio is encoded at a normalized loudness setting and the processor is programmed to:

determine a normalized loudness of the dialog; and

re-encode the asset at a second loudness setting corresponding to the normalized determined loudness, if the first loudness setting and the normalized determined loudness are different by more than a predetermined amount.

53.   A method of encoding audio of a program, comprising:

receiving a program, the program having audio encoded at a first loudness setting;

identifying dialog of the program;

determining a loudness of the dialog;

comparing the determined loudness to the first loudness setting; and

encoding the program for storage at the second loudness setting, if the first loudness setting and the second loudness are different by more than a predetermined amount.

54.   The method of claim 53, comprising identifying the dialog by:

dividing the audio into time intervals as the audio is received;

determining a loudness of each time interval as the interval is divided; and

identifying time intervals with intermediate loudness after at least a portion of the audio of the entire program is received.

55.   The method of claim 53, comprising:

determining the loudness of each time interval based on psychoacoustic

criteria.

56.   The method of claim 55, comprising:

determining the loudness of each time interval based on Leq (A).

57.   The method of claim 53, further comprising:

normalizing the determined loudness of the time intervals having

intermediate loudnesses.

58.   The method of claim 57, wherein the normalized loudness setting is a

DIALNORM setting, the method comprising determining the normalized loudness by:

determining a DIALNORM of the time intervals having intermediate

loudnesses.

59.   A system for encoding audio of a program, comprising:

a receiver to receive a program, the program having audio encoded at a

first loudness setting; and

a processor programmed to:

identify dialog of the program;

determine a loudness of the dialog;

compare the determined loudness to the first loudness setting; and

encode the program for storage at a second loudness setting

corresponding to the second loudness, if the first loudness setting and the determined

loudness are different by more than a predetermined amount.

60.   A method of encoding audio of a program, comprising:

retrieving a stored program, the program comprising audio;

identifying dialog of the audio;

determining a loudness of the dialog; and

encoding the program at a loudness setting corresponding to the

determined loudness.

61.   The method of claim 60, comprising identifying the dialog by:

dividing the audio into time intervals;

determining a loudness of each time interval;

identifying time intervals with intermediate loudnesses; and

determining a loudness of the time intervals with intermediate loudnesses.

62.   The method of claim 60, further comprising:

transmitting the program with the encoded loudness setting.

63.   The method of claim 60, comprising:

determining the loudness of each time interval based on psychoacoustic

criteria.

64.   The method of claim 63, comprising:

determining the loudness of each time interval based on Leq (A).

65.   The method of claim 60, further comprising:

normalizing the determined loudness of the time intervals having

intermediate loudnesses.

66.   The method of claim 65, comprising determining the normalized loudness

by:

determining a DIALNORM of the intervals having intermediate

loudnesses.

67.   The method of claim 60, further comprising:

determining a compression value for the audio

68.   A system for encoding audio of a program, comprising:

memory to store the program, the program comprising audio; and

a processor programmed to:

retrieve the stored program;

identify dialog of the audio;

determine a loudness of the dialog; and

encode the program at a loudness setting corresponding to the

determined loudness.

69.   The system of claim 68, wherein:

the processor is programmed to identify dialog by:

dividing the audio into time intervals;

determining a loudness of each time interval; and

identifying time intervals with intermediate loudnesses; and

the processor is programmed to determine the loudness of the dialog by:

determining a loudness of the time intervals with intermediate

loudnesses.

70.   The system of claim 68, further comprising:

a transmitter coupled to the processor, to transmit the program with the

encoded loudness setting.

71.   The system of claim 68, wherein the processor is further programmed to:

determine a compression value for the audio.